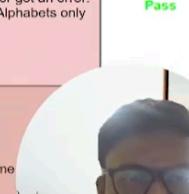




TC_002	Functionality	Validate the Take your Hair Test button on the navbar	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://fraya.health/ 2. Click on the Take the Hair Test button on the navbar 3. Verify the URL https://form.fraya.health/ after clicking on the button	NULL	The User should be redirected to https://form.fraya.health/ after clicking on the button	The User is redirected to https://form.fraya.health/ after clicking on the button	Pass
TC_003	Functionality	Validate Name Field on the Test Form With Invalid data Numeric	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://fraya.health/ 2. Click on the Take the Hair Test Button From the navbar or hero image 3. Put a Numeric data in name field and press Enter	Name: 8232	The User should get an error: Enter Alphabets only	The User got an error: Enter Alphabets only	Pass
TC_004	Functionality	Validate Name Field on the Test Form With Invalid Name	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://fraya.health/ 2. Click on the Take your Hair Test Button From the navbar or hero image	Name: S	The name should not be submitted and get an error: Put a valid name	The name	



Test Cases	TC_005	Functionality	Validate Name Field on the Test Form With valid Name	Any Operating system with a running Internet connection and Chrome Browser	2. Click on the Take your Hair Test Button from the navbar or hero image 3. Fill the name field with given data and press Enter	Name: Sukhmani	The name should be submitted	The name is submitted
	TC_006	Functionality	Validate Phone Field on the Test Form With valid Number	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://traya.health/ 2. Click on the Take your Hair Test Button from the navbar or hero image 3. Fill the name field with given data and press Enter 4. Fill the Phone number field with the given data and press Enter	Name: Sukhmani Phone Number: 7383838312	The phone number should be submitted and user should be redirected to step 4 <input type="checkbox"/> Copy <input checked="" type="checkbox"/> Sort range <input checked="" type="checkbox"/> Get link to this range	The phone number is submitted
	TC_007	Functionality	Validate Phone Field on the Test Form With Invalid Number below 10 digit	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://traya.health/ 2. Click on the Take your Hair Test Button from the navbar or hero image 3. Fill the name field with given data and press Enter	Name: Sukhmani Phone Number: 737383	The phone number should not be submitted and user should get an error: Hmm... that phone number isn't valid	The phone number is not submitted and an error message is displayed

TC_008	Functionality	Validate Phone Field on the Test Form With Invalid Number	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://traya.health/ 2. Click on the Take your Hair Test Button from the navbar or hero image 3. Fill the name field with given data and press Enter 4. Fill the Phone number field with the given data and press Enter	Name: Sukhmani Phone Number: 1232214214	The phone number should not be submitted and user should be get a error. Hmm... that phone number isn't valid	The phone number is submitted and user is redirected to the next step	
TC_009	Functionality	Validate Email Field on the Test Form With Invalid Email format	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://traya.health/ 2. Click on the Take your Hair Test Button from the navbar or hero image 3. Fill the name field with given data and press Enter 4. Fill the Phone number field with the given data and press Enter	Name: Sukhmani Phone Number: 1232214214 Email: hello@ht.ti	The Email should not be submitted and user should be get a error. Hmm... that email doesn't look valid	The Email is submitted and an error. The email does	
TC_010	Functionality	Validate Email Field on the Test Form With Invalid Email fake	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://traya.health/ 2. Click on the Take your Hair Test Button from the navbar or hero image and press Enter 3. Fill the name field with given data and press Enter 4. Fill the Phone number field with the given data and press Enter 5. Fill the Email field with the given data and press Enter	Name: Sukhmani Phone Number: 1232214214 Email: hello@ht.ti	The Email should not be submitted and should be verified	The Email is submitted and is not verified	
TC_011	Functionality	Validate Email Field on the Test Form With valid email	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://traya.health/ 2. Click on the Take your Hair Test Button from the navbar or hero image and press Enter 3. Fill the name field with given data and press Enter 4. Fill the Phone number field with the given data and press Enter 5. Fill the Email field with the given data and press Enter	Name: Sukhmani Phone Number: 1232214214 Email: sukhmani.dev006@gmail.com	The Email should be submitted and should be verified	The Email is submitted and is not verified	
TC_020	Performance	Check the response time of the website with single request	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://traya.health/ 2. Right click and go to inspect and go to networks tab 3. Click on the Take your Hair Test Button from the navbar or hero image and press Enter 4. Fill required data in input fields 5. Measure the response time of every request	Name: Sukhmani Phone Number: 1232214214 Email: sukhmani.dev006@gmail.com Age: 25	The server should be stable and give constant results	The server is stable and giving constant results	
TC_021	Performance	Check the response time of the website with multiple requests	Any Operating system with a running Internet connection and Chrome Browser	1. Visit the URL On the browser https://traya.health/ 2. Right click and go to inspect and go to networks tab 3. Click on the Take your Hair Test Button from the navbar or hero image and press Enter 4. Fill required data in	Name: Sukhmani Phone Number: 1232214214 Email: sukhmani.dev006@gmail.com	The server should be stable and give constant results	The server giving con	

TC_023	Compatibility	Checking Browser compatibility by testing on different browsers	Any Operating system with a running Internet connection and Edge Browser	<p>1. Visit the URL On the browser https://trava.health/</p> <p>2. Click on the Take your Hair Test Button from the navbar or hero image and press Enter</p> <p>3. Fill the name field with given data and press Enter</p> <p>4. Fill the Phone number field with the given data and press Enter</p> <p>5. Fill the Email field with the given data and press Enter</p> <p>6. Fill the age field with given data and press enter</p>	<p>Name: Sukhmani Phone Number: 9898498545 Email: sukhmani.dev006@gmail.com Age: 25</p>	The data should be submitted and user should be redirected to next steps	The data is submitted and user is redirected to next steps	Pass
TC_019	Functionality	Validate Age input field with valid age	Any Operating system with a running Internet connection and Chrome Browser	<p>1. Visit the URL On the browser https://trava.health/</p> <p>2. Click on the Take your Hair Test Button from the navbar or hero image and press Enter</p> <p>3. Fill the name field with given data and press Enter</p> <p>4. Fill the Phone number field with the given data and press Enter</p> <p>5. Fill the Email field with the given data and press Enter</p> <p>6. Fill the age field with given data and press enter</p>	<p>Name: Sukhmani Phone Number: 1232214214 Email: sukhmani.dev006@gmail.com Age: 25</p>	The age should be submitted	The age is submitted	Pass

Test Case Creation, Experience-Based Test Design Technique

masai



Test Case Creation

To ensure comprehensive, consistent, and repeatable testing.

Experience-Based Testing

To fill gaps in documentation, find subtle bugs, and leverage expertise.



Test Case Creation

Writing structured test cases with steps, data, and expected outcomes.

Experience-Based Testing

Testing using intuition, error prediction, and real-time exploration.



Introduction

Experience-based test design techniques leverage the tester's past experience, intuition, and knowledge to identify defects. These techniques complement structured methods by addressing gaps in requirement documentation and discovering subtle issues.

Together, test case creation and experience-based techniques form a balanced approach for comprehensive software testing.

Test case creation + experience based testing = balanced approach for effective testing

Key Concepts and Definitions

Test Case

A test case is a structured document containing inputs, execution conditions, and expected results for testing a specific functionality of a system.

Key Components of a Test Case:

Test Case ID: A unique identifier for tracking.

Test Scenario: A high-level idea of what is being tested.

Test Steps: Sequential actions required to execute the test.

Test Data: Inputs required to carry out the test.

Expected Result: The predicted behavior of the system under test.

Actual Result: The observed behavior during execution (filled in later).

Pass/Fail Status: Indicates whether the test case was successful.

Let's understand the analogy of Experience Based Test Design Technique

Chef's Role:

Just as a chef relies on their experience and intuition to adjust spices, ingredients, and presentation, a tester uses their knowledge of the system, past defects, and domain expertise to find potential issues in the software.



When to go for Experience based Test case design Technique?

When there is no requirement

When there are requirements but are unclear

When there are requirements but complex to understand(Exploratory Testing)

Exploratory Testing is a part of Experience based Test case design technique

Types of Experience-Based Techniques**1. Error Guessing**

Error guessing is an experience-based software testing technique where testers anticipate possible errors based on their knowledge, intuition, and previous experience with similar systems. Unlike structured approaches (e.g., boundary value analysis or equivalence partitioning), error guessing relies on creativity and expertise to identify defects.

Why Use Error Guessing?

- To detect defects missed by formal testing techniques.
- To address undocumented edge cases or unusual scenarios.
- To test "what could go wrong" based on past patterns of failure.

**Example 1: Login Form**

User enters an empty username or password.

Username exceeds the maximum allowed length.

Password contains unsupported special characters.

Test Cases:

Input: Username = null, Password = password123.

Input: Username =aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
(50+ characters).

Input: Username = admin, Password = !@#\$%^&*().

Best Practices for Error Guessing**Focus on High-Risk Areas:**

Concentrate on areas where failure could have severe consequences (e.g., payment processing).

Use Checklists:

Maintain a checklist of common error types (e.g., empty fields, invalid characters, boundary inputs).

Combine with Other Techniques:

Use error guessing alongside formal testing techniques for comprehensive coverage.

2.Exploratory testing

It is a hands-on, interactive approach to software testing where testers simultaneously learn about the system, design tests, and execute them in real time. It emphasizes creativity, intuition, and adaptability rather than following pre-defined test cases.

Unlike scripted testing, which relies on predefined test steps and outcomes, exploratory testing allows testers to dynamically explore the application and adapt their approach based on observed behaviors. This method is particularly effective in identifying defects that structured techniques may miss.

Characteristics of Exploratory Testing

Unscripted: No predefined steps or expected results; testers design and execute tests on the fly.

Adaptive: Changes focus based on system behavior and observed defects.

Intuition-Driven: Relies on the tester's experience, creativity, and understanding of the application.

When to Use Exploratory Testing?

When requirements or documentation are incomplete or ambiguous.

During initial testing phases to quickly understand the application's behavior.

To supplement structured testing techniques for uncovering subtle defects.

In Agile or rapid development cycles where time for formal test planning is limited.

Examples of Exploratory Testing

1. Login Feature

Test valid and invalid login credentials.

Leave the password field empty and submit.

Enter an email address as the username to check error handling.

2. E-commerce Checkout

Add items to the cart, proceed to checkout, and close the browser midway.

Apply expired or invalid promo codes to test error handling.

Test concurrent orders from two different accounts using the same inventory item.

3. Checklist-based testing

It is a systematic approach to software testing where testers validate functionality using a predefined list of test items or scenarios. This list ensures that all critical aspects of the application are covered during testing, reducing the likelihood of missing defects.

Checklists are typically created based on:

Requirements or specifications.

Common defect patterns.

Domain knowledge and previous experiences.

This method is especially useful in **Agile environments** or situations where time is limited, as it provides a quick and efficient way to verify the application's functionality.

3. Session-Based Testing

Session-Based Testing (SBT) is a structured yet exploratory testing approach that divides the testing process into focused, time-boxed sessions. This method aims to combine the flexibility of exploratory testing with a clear framework for tracking and reporting results.

Key Features of Session-Based Testing

Time-Boxed Sessions:

Testing is performed in predefined blocks of time, typically lasting 60–120 minutes.

Charters:

Each session is guided by a **test charter**, which outlines the specific focus area or goal for the session.

Simultaneous Test Design and Execution:

Testers create and execute test cases in real-time based on the application's behavior and the defined charter.

Session Reports:

Detailed notes and observations are recorded during the session to document the outcomes, issues, and next steps.

Accountability:

Sessions are reviewed by leads or managers to ensure alignment with project goals and priorities.

Example of a Session-Based Test Charter

Charter:

“Test the checkout functionality of an e-commerce application.”

Session Details:

Duration: 90 minutes

Focus Areas:

Valid and invalid payment methods.

Coupon code application.

Address validation.

4. Ad-Hoc Testing

Ad-Hoc Testing is an informal and unstructured testing method where the tester explores the application without predefined test cases or documentation. The goal is to identify defects by exploring the software randomly, leveraging the tester's intuition, creativity, and domain knowledge.

Adhoc Testing - It is also called as Random Testing/Monkey Testing/Out of box Testing/Negative Testing

Test account number

12 digits

Key Features of Ad-Hoc Testing

No Predefined Plan:

The testing process is unscripted, with no prior test case preparation.

Exploratory in Nature:

- Relies on the tester's knowledge, experience, and understanding of the system to identify defects.

Random and Unpredictable:

Testers intentionally attempt unusual or extreme scenarios to uncover bugs.

Examples of Ad-Hoc Testing Scenarios

Scenario: Testing an e-commerce website

Try adding the same item to the cart repeatedly to see if the cart updates correctly.

Enter special characters in the search bar to check how the system handles unexpected inputs.

Test checkout with an expired credit card.

Scenario: Testing a login module

Enter multiple incorrect passwords to check for account lockout behavior.

Leave mandatory fields blank and submit the form.